PyMOL basics – a small example

By Thomas Holberg Blicher

Getting started

In Windows and Mac OS X, click the PyMOL icon. If you are using UNIX/Linux, type *pymol* at the prompt. In windows and UNIX/Linux, PyMOL starts with two windows: the Viewer window, where the structures are displayed, and the External (Tcl/Tk) GUI window, which has a collection of pull-down menus and a command input field. In Mac OS X the two windows are fused but for most purposes behave the same. This tutorial is intended for users who don't use PyMOL every day, and therefore menus will be used as much as possible, to reduce the amount of commands that needs to be remembered by the occasional user. If you are going to be using PyMOL a lot, it is worthwhile to get hold of the manual (see links at the end of the tutorial) and learn the commands. You can ease your work considerably through scripting.

We will start the actual tutorial by working on the PDB file 1duz. To download the file directly into PyMOL, type the following command at the command line:

fetch 1duz

This will download the file to PyMOL's default directory. You can also download the file from www.rcsb.org/pdb, to have more control of where the file is saved. The fetch command also checks the default directory to see if the file has been downloaded before.

First, take a look at the PDB file itself. PDB files are text files with a header containing various information about the structure followed by the actual coordinates of the atoms in the structure. In the header, you can find information on how to generate the biomolecule from the coordinates in the PDB file in the REMARK 300 and REMARK 350 fields:

```
REMARK 300
REMARK 300 BIOMOLECULE: 1, 2
REMARK 300 THIS ENTRY CONTAINS THE CRYSTALLOGRAPHIC ASYMMETRIC UNIT
REMARK 300 WHICH CONSISTS OF 6 CHAIN(S). SEE REMARK 350 FOR
REMARK 300 INFORMATION ON GENERATING THE BIOLOGICAL MOLECULE(S).
REMARK 350
REMARK 350 GENERATING THE BIOMOLECULE
REMARK 350 COORDINATES FOR A COMPLETE MULTIMER REPRESENTING THE KNOWN
REMARK 350 BIOLOGICALLY SIGNIFICANT OLIGOMERIZATION STATE OF THE
REMARK 350 MOLECULE CAN BE GENERATED BY APPLYING BIOMT TRANSFORMATIONS
REMARK 350 GIVEN BELOW.  BOTH NON-CRYSTALLOGRAPHIC AND
REMARK 350 CRYSTALLOGRAPHIC OPERATIONS ARE GIVEN.
REMARK 350
REMARK 350 BIOMOLECULE: 1
REMARK 350 APPLY THE FOLLOWING TO CHAINS: A, B, C
REMARK 350   BIOMT1   1  1.000000  0.000000  0.000000        0.00000
REMARK 350   BIOMT2   1  0.000000  1.000000  0.000000        0.00000
REMARK 350   BIOMT3   1  0.000000  0.000000  1.000000        0.00000
REMARK 350 BIOMOLECULE: 2
REMARK 350 APPLY THE FOLLOWING TO CHAINS: D, E, F
REMARK 350   BIOMT1   2  1.000000  0.000000  0.000000        0.00000
REMARK 350   BIOMT2   2  0.000000  1.000000  0.000000        0.00000
REMARK 350   BIOMT3   2  0.000000  0.000000  1.000000        0.00000
```

In this case, the PDB file contains 6 chains, where the A, B and C chains make up the first biomolecule, and the D, E and F chains make up the second. A bit further down in the header, you can find the SEQRES fields, which gives the sequence of the chains:

```
SEQRES   1 A  275  GLY SER HIS SER MET ARG TYR PHE PHE THR SER VAL SER
SEQRES   2 A  275  ARG PRO GLY ARG GLY GLU PRO ARG PHE ILE ALA VAL GLY
SEQRES   3 A  275  TYR VAL ASP ASP THR GLN PHE VAL ARG PHE ASP SER ASP
```

```
SEQRES   4 A  275  ALA ALA SER GLN ARG MET GLU PRO ARG ALA PRO TRP ILE
SEQRES   5 A  275  GLU GLN GLU GLY PRO GLU TYR TRP ASP GLY GLU THR ARG
SEQRES   6 A  275  LYS VAL LYS ALA HIS SER GLN THR HIS ARG VAL ASP LEU
SEQRES   7 A  275  GLY THR LEU ARG GLY TYR TYR ASN GLN SER GLU ALA GLY
SEQRES   8 A  275  SER HIS THR VAL GLN ARG MET TYR GLY CYS ASP VAL GLY
SEQRES   9 A  275  SER ASP TRP ARG PHE LEU ARG GLY TYR HIS GLN TYR ALA
SEQRES  10 A  275  TYR ASP GLY LYS ASP TYR ILE ALA LEU LYS GLU ASP LEU
SEQRES  11 A  275  ARG SER TRP THR ALA ALA ASP MET ALA ALA GLN THR THR
SEQRES  12 A  275  LYS HIS LYS TRP GLU ALA ALA HIS VAL ALA GLU GLN LEU
SEQRES  13 A  275  ARG ALA TYR LEU GLU GLY THR CYS VAL GLU TRP LEU ARG
SEQRES  14 A  275  ARG TYR LEU GLU ASN GLY LYS GLU THR LEU GLN ARG THR
SEQRES  15 A  275  ASP ALA PRO LYS THR HIS MET THR HIS HIS ALA VAL SER
SEQRES  16 A  275  ASP HIS GLU ALA THR LEU ARG CYS TRP ALA LEU SER PHE
SEQRES  17 A  275  TYR PRO ALA GLU ILE THR LEU THR TRP GLN ARG ASP GLY
SEQRES  18 A  275  GLU ASP GLN THR GLN ASP THR GLU LEU VAL GLU THR ARG
SEQRES  19 A  275  PRO ALA GLY ASP GLY THR PHE GLN LYS TRP ALA ALA VAL
SEQRES  20 A  275  VAL VAL PRO SER GLY GLN GLU GLN ARG TYR THR CYS HIS
SEQRES  21 A  275  VAL GLN HIS GLU GLY LEU PRO LYS PRO LEU THR LEU ARG
SEQRES  22 A  275  TRP GLU
SEQRES   1 B  100  MET ILE GLN ARG THR PRO LYS ILE GLN VAL TYR SER ARG
SEQRES   2 B  100  HIS PRO ALA GLU ASN GLY LYS SER ASN PHE LEU ASN CYS
SEQRES   3 B  100  TYR VAL SER GLY PHE HIS PRO SER ASP ILE GLU VAL ASP
SEQRES   4 B  100  LEU LEU LYS ASN GLY GLU ARG ILE GLU LYS VAL GLU HIS
SEQRES   5 B  100  SER ASP LEU SER PHE SER LYS ASP TRP SER PHE TYR LEU
SEQRES   6 B  100  LEU TYR TYR THR GLU PHE THR PRO THR GLU LYS ASP GLU
SEQRES   7 B  100  TYR ALA CYS ARG VAL ASN HIS VAL THR LEU SER GLN PRO
SEQRES   8 B  100  LYS ILE VAL LYS TRP ASP ARG ASP MET
SEQRES   1 C    9  LEU LEU PHE GLY TYR PRO VAL TYR VAL
SEQRES   1 D  275  GLY SER HIS SER MET ARG TYR PHE PHE THR SER VAL SER
SEQRES   2 D  275  ARG PRO GLY ARG GLY GLU PRO ARG PHE ILE ALA VAL GLY
SEQRES   3 D  275  TYR VAL ASP ASP THR GLN PHE VAL ARG PHE ASP SER ASP
SEQRES   4 D  275  ALA ALA SER GLN ARG MET GLU PRO ARG ALA PRO TRP ILE
SEQRES   5 D  275  GLU GLN GLU GLY PRO GLU TYR TRP ASP GLY GLU THR ARG
SEQRES   6 D  275  LYS VAL LYS ALA HIS SER GLN THR HIS ARG VAL ASP LEU
SEQRES   7 D  275  GLY THR LEU ARG GLY TYR TYR ASN GLN SER GLU ALA GLY
SEQRES   8 D  275  SER HIS THR VAL GLN ARG MET TYR GLY CYS ASP VAL GLY
SEQRES   9 D  275  SER ASP TRP ARG PHE LEU ARG GLY TYR HIS GLN TYR ALA
SEQRES  10 D  275  TYR ASP GLY LYS ASP TYR ILE ALA LEU LYS GLU ASP LEU
SEQRES  11 D  275  ARG SER TRP THR ALA ALA ASP MET ALA ALA GLN THR THR
SEQRES  12 D  275  LYS HIS LYS TRP GLU ALA ALA HIS VAL ALA GLU GLN LEU
SEQRES  13 D  275  ARG ALA TYR LEU GLU GLY THR CYS VAL GLU TRP LEU ARG
SEQRES  14 D  275  ARG TYR LEU GLU ASN GLY LYS GLU THR LEU GLN ARG THR
SEQRES  15 D  275  ASP ALA PRO LYS THR HIS MET THR HIS HIS ALA VAL SER
SEQRES  16 D  275  ASP HIS GLU ALA THR LEU ARG CYS TRP ALA LEU SER PHE
SEQRES  17 D  275  TYR PRO ALA GLU ILE THR LEU THR TRP GLN ARG ASP GLY
SEQRES  18 D  275  GLU ASP GLN THR GLN ASP THR GLU LEU VAL GLU THR ARG
SEQRES  19 D  275  PRO ALA GLY ASP GLY THR PHE GLN LYS TRP ALA ALA VAL
SEQRES  20 D  275  VAL VAL PRO SER GLY GLN GLU GLN ARG TYR THR CYS HIS
SEQRES  21 D  275  VAL GLN HIS GLU GLY LEU PRO LYS PRO LEU THR LEU ARG
SEQRES  22 D  275  TRP GLU
SEQRES   1 E  100  MET ILE GLN ARG THR PRO LYS ILE GLN VAL TYR SER ARG
SEQRES   2 E  100  HIS PRO ALA GLU ASN GLY LYS SER ASN PHE LEU ASN CYS
SEQRES   3 E  100  TYR VAL SER GLY PHE HIS PRO SER ASP ILE GLU VAL ASP
SEQRES   4 E  100  LEU LEU LYS ASN GLY GLU ARG ILE GLU LYS VAL GLU HIS
SEQRES   5 E  100  SER ASP LEU SER PHE SER LYS ASP TRP SER PHE TYR LEU
SEQRES   6 E  100  LEU TYR TYR THR GLU PHE THR PRO THR GLU LYS ASP GLU
SEQRES   7 E  100  TYR ALA CYS ARG VAL ASN HIS VAL THR LEU SER GLN PRO
SEQRES   8 E  100  LYS ILE VAL LYS TRP ASP ARG ASP MET
SEQRES   1 F    9  LEU LEU PHE GLY TYR PRO VAL TYR VAL
```

The molecule we are looking at is an MHC molecule. (Molecule of the month at the PDB in February 2005: http://www.rcsb.org/pdb/molecules/pdb62_1.html.) MHC molecules consist of two protein chains, here A and B (and D and E) and a peptide, here C (and F), which in this case is 9 residues long. When you have the PDB file in your PyMOL working directory, open it by selecting "open" under the "File" menu. Now the structure is shown in the Viewer window in the "lines" representation. You can now see the two biomolecules plus a lot of water molecules (red crosses). We only want to look at one molecule, so we define an object consisting of chains A-C, by entering at the command prompt:

create molecule1, chain A or chain B or chain C

This creates an object named "molecule1" consisting of one molecule only, which now shows up under "1duz" on the list of objects in the viewer window. (Note, that the PyMOL language is case-sensitive, but upper case is not used for commands.) Next to the object names in the Viewer window are five buttons: A(ctions), S(how), H(ide), L(abels) and C(olor). Hide the 1duz object now by selecting H(ide) everything. If the "molecule1" object is not centred, choose "zoom" under A(ction). This puts your selected object in the centre of the display and makes it fill out the whole window. S(how) the "molecule1" object in different representations, try them all to see what they do. Note, that you have to H(ide) the representations after use, or they will all be shown on top of each other. When you have seen them all, go back to the "cartoon" representation. Now experiment with C(olor) until you have found something you are pleased with. Rotate the molecule using the left mouse button and zoom (move in the Z direction) using the right mouse button until you are also pleased with the orientation of the molecule. There is a table with an overview of the different mouse functions in the lower right-hand corner of the Viewer window. Now you are ready to create a publication-quality image. It's dead easy - just press "Ray" in the GUI window (or type "ray" at the prompt). This will take a few seconds to a few minutes, depending on the size and complexity of the image you want to ray-trace. To ray-trace an image with dimensions different from those of the viewer window type

ray x, y

Here x and y are the width and height of the image in pixels.

Changing settings

If you choose "Edit all" under "Settings" in the GUI window, you can see all the settings that PyMOL allows you to change. If you like slimmer sticks, then change the corresponding value in PyMOL (stick_radius). The background color can be changed to any RGB color by changing "bg_rgb". The three numbers from 0-1 represent the amount of red, green and blue, respectively. Try changing the background color to red. Typing the command below is a shortcut using already named colours (see the C(olor) menu on the right of viewer window).

bg white

Saving

To save your image, choose "Save Image" in the "File" menu in the GUI window. If you have to leave PyMOL now, but expect to come back and work some more on your image later, you can save your PyMOL session, so you won't have to start all over again next time. Do this by selecting "Save session" or "Save session as" under the "File" menu. It is a good idea to save your session before doing something radical, in this way it works as an "undo" button. Please note that except for a few specialized cases, PyMOL does not have an undo function.

Selections

This example has demonstrated many of the most often used features in PyMOL, but in order to create more complex figures, you will need to know a little more about selections.

If you wish to do something with just a subset of the atoms in an object, you can create either a named selection or a new object consisting of the subset you are interested in. Selections and objects play slightly

different roles in PyMOL. For most purposes, I recommend creating new objects with the selections you wish to work with. Objects are created by the command "create", which has the syntax

create name, selection

where ***name*** = object to create (or modify) and ***selection*** = atoms to include in the object

To understand how selections work, let's have another look at the PDB file. After the header records, the actual coordinates of the atoms are listed in the following format:

```
ATOM      1  N   GLY A   1      14.752  -6.145  13.692  1.00 25.90           N
ATOM      2  CA  GLY A   1      15.556  -5.512  12.629  1.00 26.24           C
ATOM      3  C   GLY A   1      15.173  -4.015  12.551  1.00 25.34           C
ATOM      4  O   GLY A   1      14.683  -3.476  13.556  1.00 26.37           O
ATOM      5  N   SER A   2      15.385  -3.415  11.390  1.00 24.50           N
ATOM      6  CA  SER A   2      15.117  -1.999  11.027  1.00 23.82           C
ATOM      7  C   SER A   2      13.621  -1.646  10.953  1.00 21.82           C
ATOM      8  O   SER A   2      12.801  -2.540  10.698  1.00 20.40           O
ATOM      9  CB  SER A   2      15.721  -1.709   9.635  1.00 24.56           C
ATOM     10  OG  SER A   2      17.138  -1.917   9.705  1.00 29.71           O
ATOM     11  N   HIS A   3      13.271  -0.356  11.200  1.00 20.13           N
ATOM     12  CA  HIS A   3      11.821   0.031  11.153  1.00 18.92           C
ATOM     13  C   HIS A   3      11.657   1.450  10.570  1.00 18.05           C
ATOM     14  O   HIS A   3      12.661   2.139  10.422  1.00 17.09           O
ATOM     15  CB  HIS A   3      11.225   0.000  12.561  1.00 18.31           C
ATOM     16  CG  HIS A   3      11.161  -1.410  13.108  1.00 17.87           C
ATOM     17  ND1 HIS A   3      10.185  -2.297  12.725  1.00 18.42           N
ATOM     18  CD2 HIS A   3      11.937  -2.071  13.989  1.00 19.01           C
ATOM     19  CE1 HIS A   3      10.362  -3.467  13.333  1.00 19.54           C
ATOM     20  NE2 HIS A   3      11.415  -3.344  14.121  1.00 19.01           N
```

In the first column is the field "ATOM" which indicates that the rest of the line contains atomic coordinate information. The next column is the atom number, then comes atom name (in PyMOL called "**name**"), then residue name (PyMOL: **resn**), chain id (PyMOL: **chain**), residue number (PyMOL: **resi**). The next three columns are the x, y and z coordinates of the atoms, followed by the occupancy (usually 1.00) and the B-factor. The last column is rarely used (and is missing in some PDB files), but gives the atom type (element).

B-factors

The B-factors can be used to illustrate flexibility of the structure (the higher the B-factor, the more flexible the structure). The structure can be coloured by B-factor, if you choose "spectrum" in the C(olor) menu, and then "b-factors". Try this. You can see that the structure is most rigid in the centre and most flexible in the loop regions.

The B-factor column is actually very useful. You can use the fact that a structure can be coloured by the numbers in the B-factor column, and replace the B-factors with values for other features that you wish to depict on your structure. This could for example be a measure of conservation, which would turn your image into a type of 3-D logo plot.

Selection syntax

The selection syntax of PyMOL will be demonstrated through the examples below. To create an object consisting of just the carbon atoms in the structure, enter

create carbons, name ca+cb+cg+cd

Try this. You will notice, that it chooses all the carbons in all the chains (including chains D-F). If you want to only choose the carbons in chains A-C, you can either delete the 1duz object, then recreate the carbons object, or you can limit your selection to chains A-C:

create carbons, (name ca+cb+cg+cd) and (chain A or chain B or chain C)

Note that if you have a (+)-separated list of identifiers, no spaces are allowed. If you want to show a range of residues, such as the first 10 residues of the N-terminal, you can use a (-):

create Nterminal, (resi 1-10) and (chain A)

(but you cannot use both (+) and (-) in the same command; *resi 1-10+36* is NOT allowed).

create bb, name c+o+n+ca

Selection algebra

You have already seen how to include residues that are either in chain A or chain B or chain C in the above example. Here is a list of other useful selection operators and modifiers:

Operator	Effect
not s1	Selects atoms that are not included in s1 (PyMOL ex: *create sidechains, not bb*)
s1 and s2	Selects atoms included in both s1 and s2
s1 or s2	Selects atoms included in either s1 or s2
s1 around X	Selects atoms with centres within X Angstroms of the centre of any atom in s1
s1 expand X	Expands s1 by all atoms within X Angstroms of the centre of any atom in s1
s1 within X of s2	Selects atoms in s1 that are within X Angstroms of s2
byres s1	Expands selection to complete residues.
byobject s1	Expands selection to complete objects.
neighbor s1	selects atoms directly bonded to s1.

Surfaces

A very useful feature of PyMOL (and most other molecular viewing programs) is the ability to generate molecular surfaces to give a more realistic impression on the shape of binding pockets, active sites, accessibility, overall shape etc. To generate a surface, simply go to S(how) menu and select "surface". This will generate a surface covering the chosen selection or object. Let us first generate a molecule to work with:

create molecule1, 1duz and (chain a or chain b or chain c)

Chain C is the bound peptide. Try the following:

select mhc, molecule1 and (chain a or chain b)

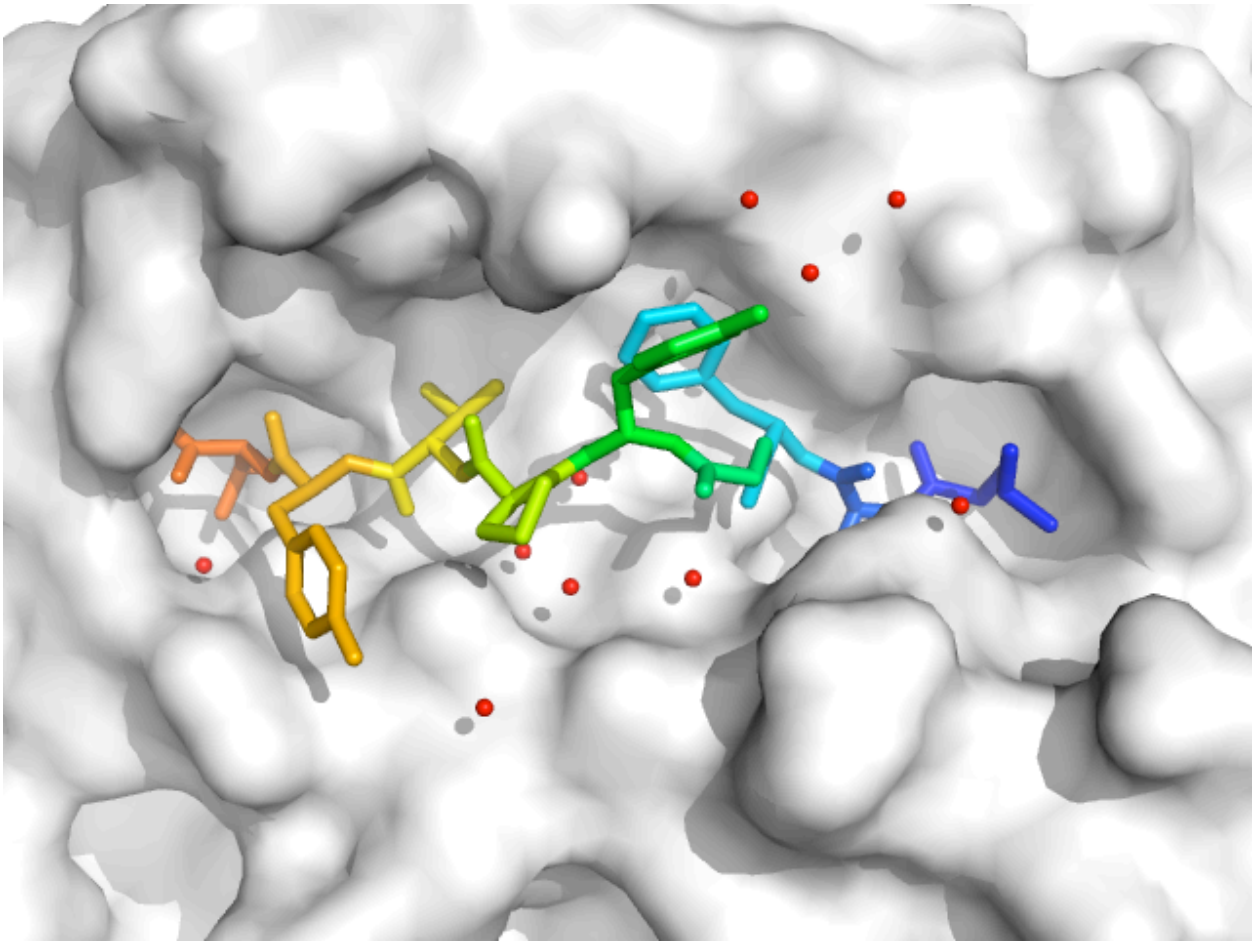
Now cover this selection with a surface. How does it look? Are you satisfied? Probably not. You will notice that the surface appears to have a hole in it where the peptide contributes to the surface of the molecule. To get a surface that covers only the MHC molecule part and not the peptide, generate a separate object consisting of only chain A and chain B:

create MHCmolecule, molecule1 and (chain a or chain b)

Now add a surface to this molecule. Better, right? Let us also make a separate object for the peptide:

create pep, molecule1 and chain C

Unselect all molecules except MHCmolecule and pep. Now try and make a figure like the one shown below: The surface is covering the A and B chains, while the peptide - the C chain - is shown in stick representation. Try first to make the image without the water molecules (small red spheres).



If that was too easy, then have a go at this one: Same as above but now with the water molecules within 4 Angstroms of the peptide included. Hint: You will need to use some of the selection examples from the Selection algebra section above.

PyMOL links

PyMOL home:
http://pymol.sourceforge.net/

PyMOL manual:
http://pymol.sourceforge.net/newman/user/toc.html

PyMOL Wiki:
http://www.pymolwiki.org/index.php/Main_Page

PyMOL settings (documented):
http://cluster.earlham.edu/detail/bazaar/software/pymol/modules/pymol/setting.py